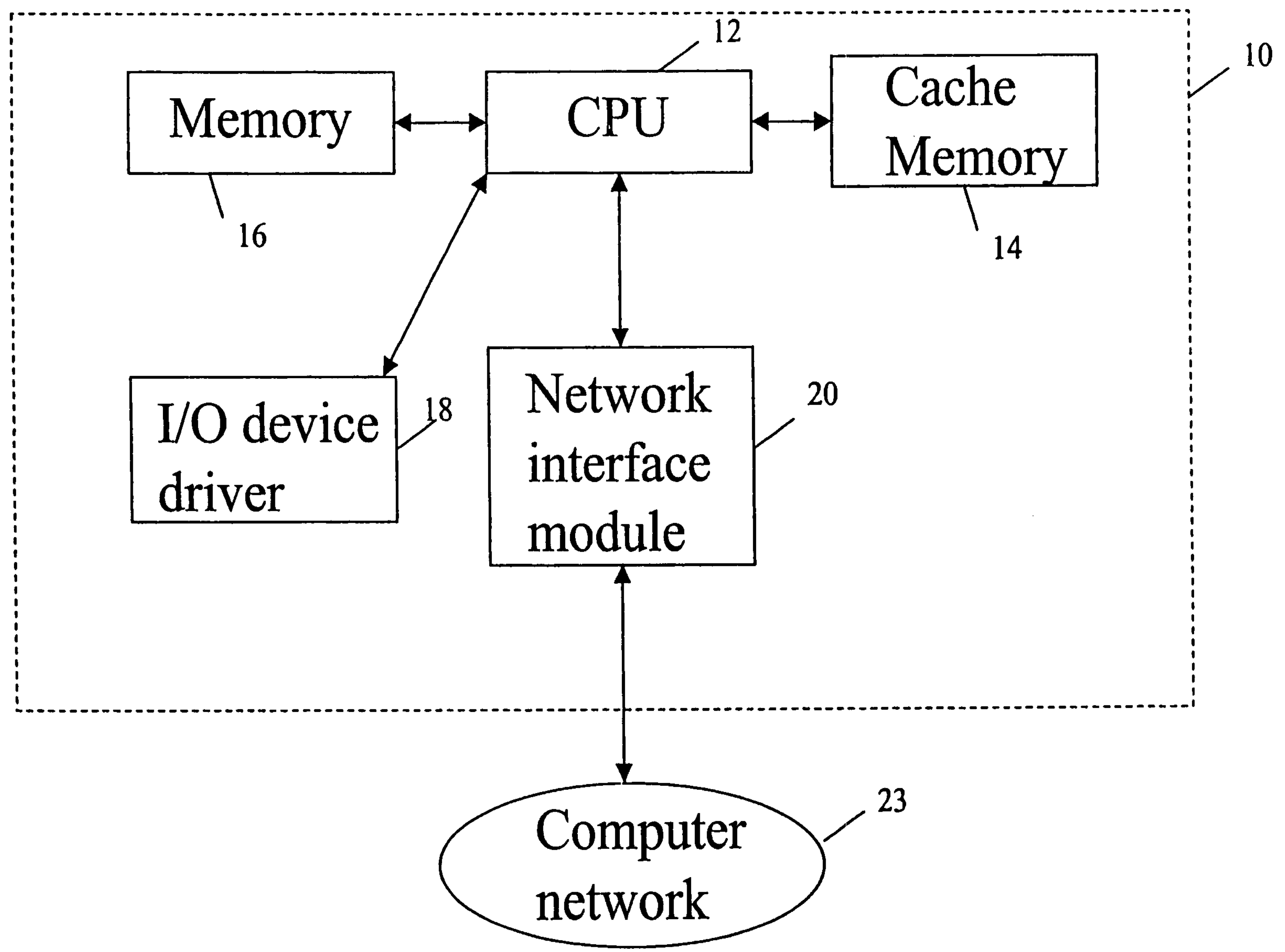

10
12
Memory
CPU
Cache Memory
16
14
I/O device driver
18
Network interface module
20
Computer network
23

FIG. 1

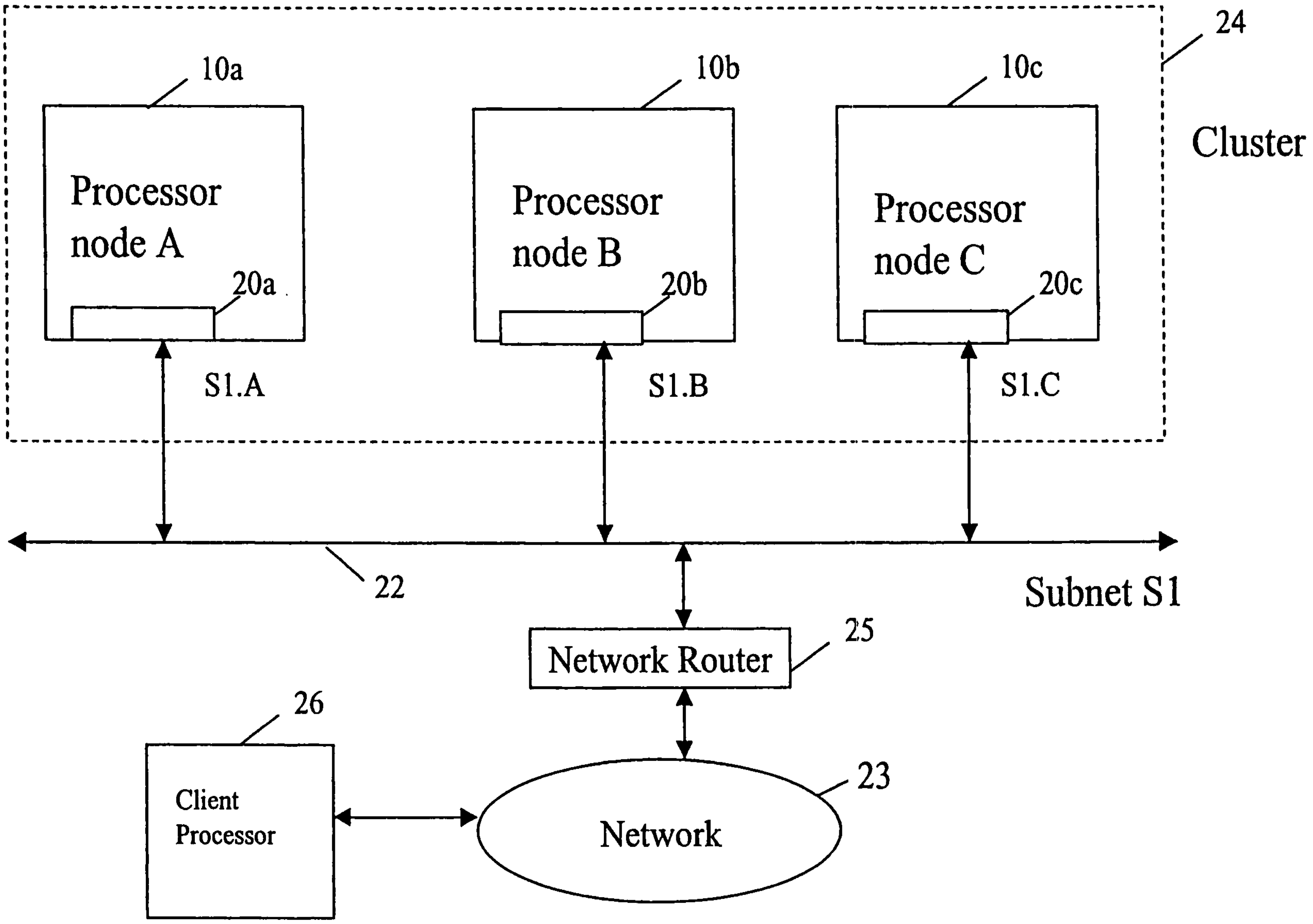
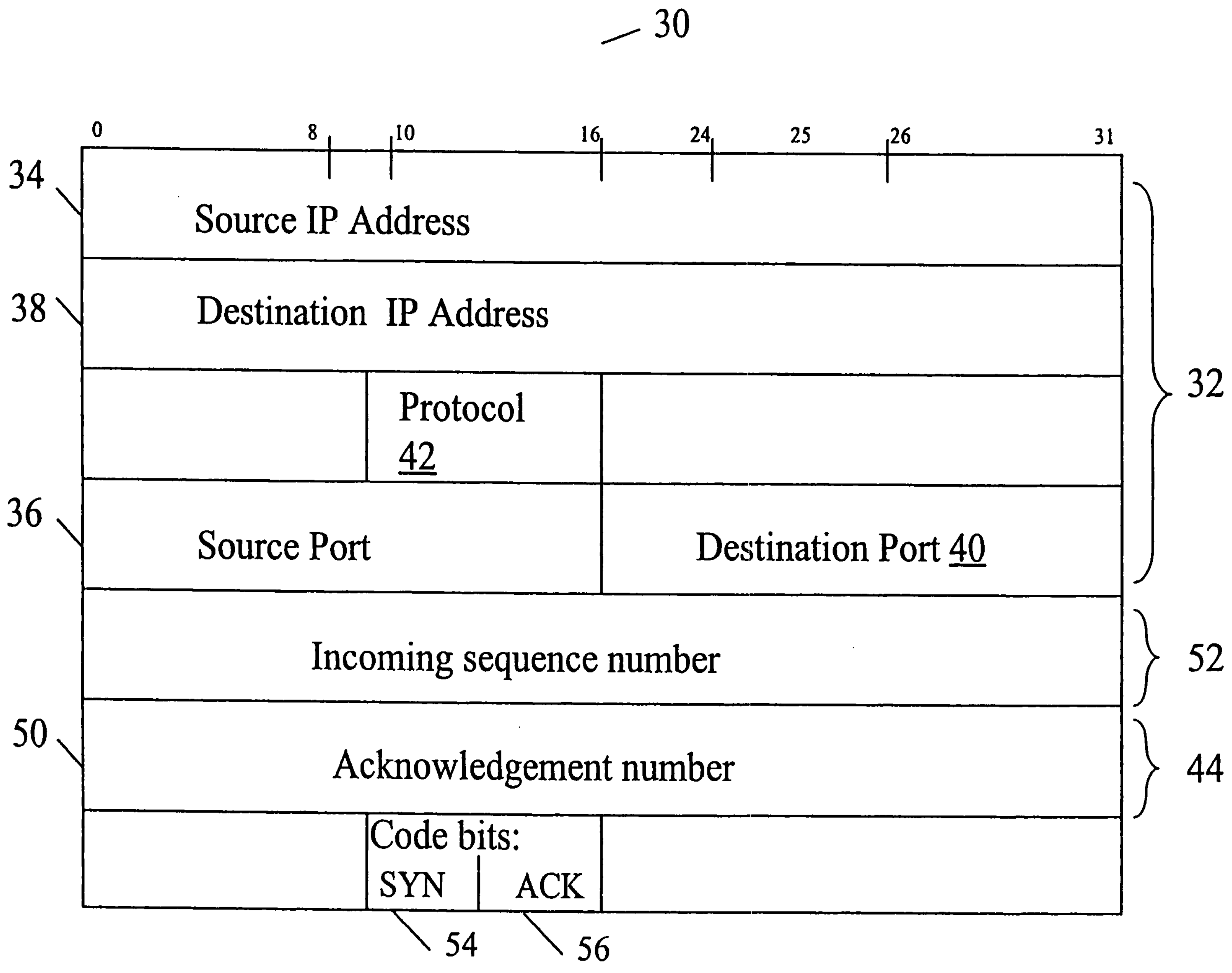

24
10a
10b
10c
Cluster
Processor node A
Processor node B
Processor node C
20a
20b
20c
S1.A
S1.B
S1.C
22
Subnet S1
25
Network Router
26
Client Processor
23
Network

FIG.2

FIG.3


Source IP Address
Destination IP Address
Protocol
Source Port
Destination Port 40
Incoming sequence number
Acknowledgement number
Code bits:
SYN
ACK

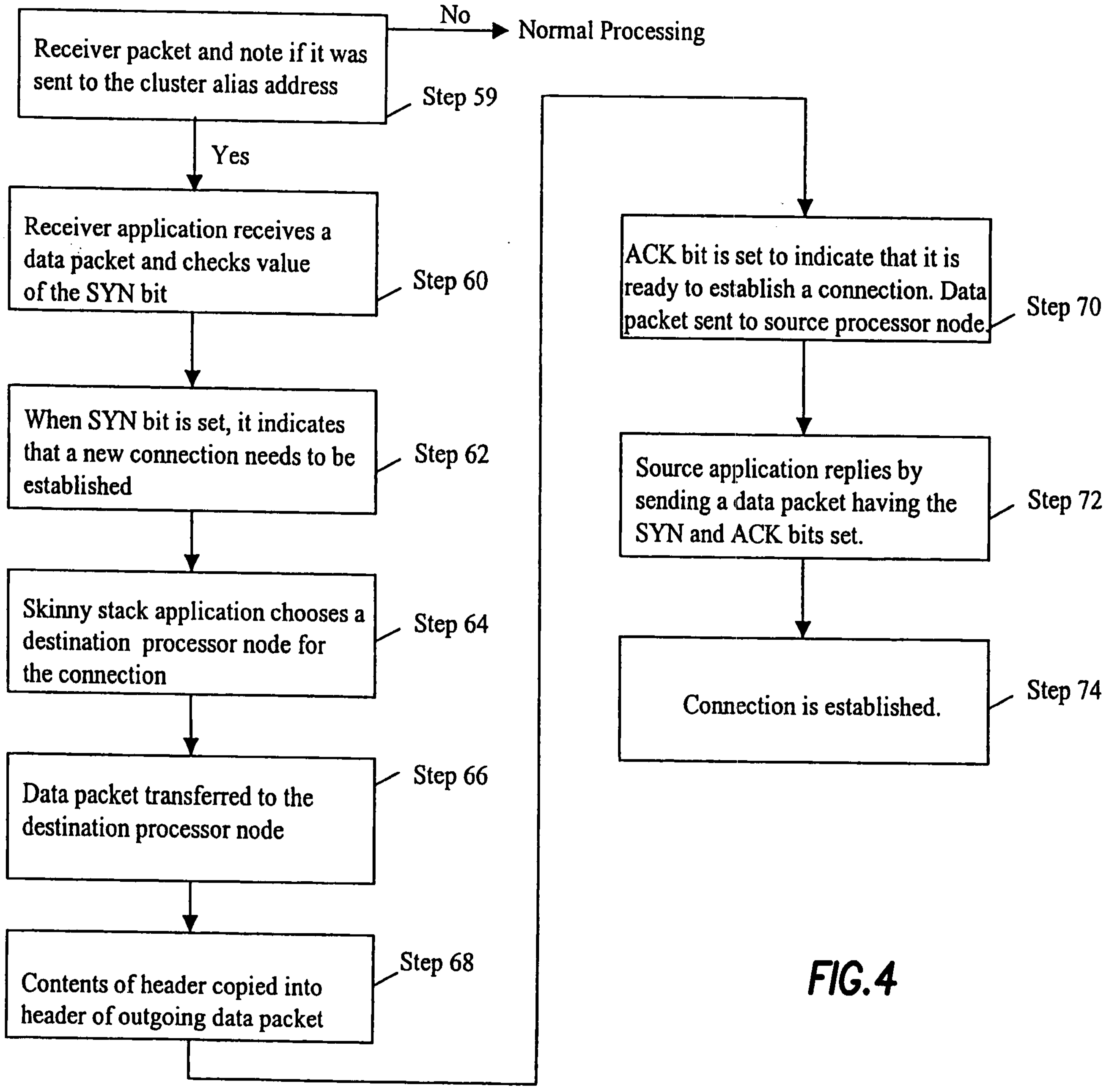

Receiver packet and note if it was sent to the cluster alias address
Step 59
No
Normal Processing
Yes
Receiver application receives a data packet and checks value of the SYN bit
Step 60
When SYN bit is set, it indicates that a new connection needs to be established
Step 62
Skinny stack application chooses a destination processor node for the connection
Step 64
Data packet transferred to the destination processor node
Step 66
Contents of header copied into header of outgoing data packet
Step 68
ACK bit is set to indicate that it is ready to establish a connection. Data packet sent to source processor node.
Step 70
Source application replies by sending a data packet having the SYN and ACK bits set.
Step 72
Connection is established.
Step 74

FIG.4

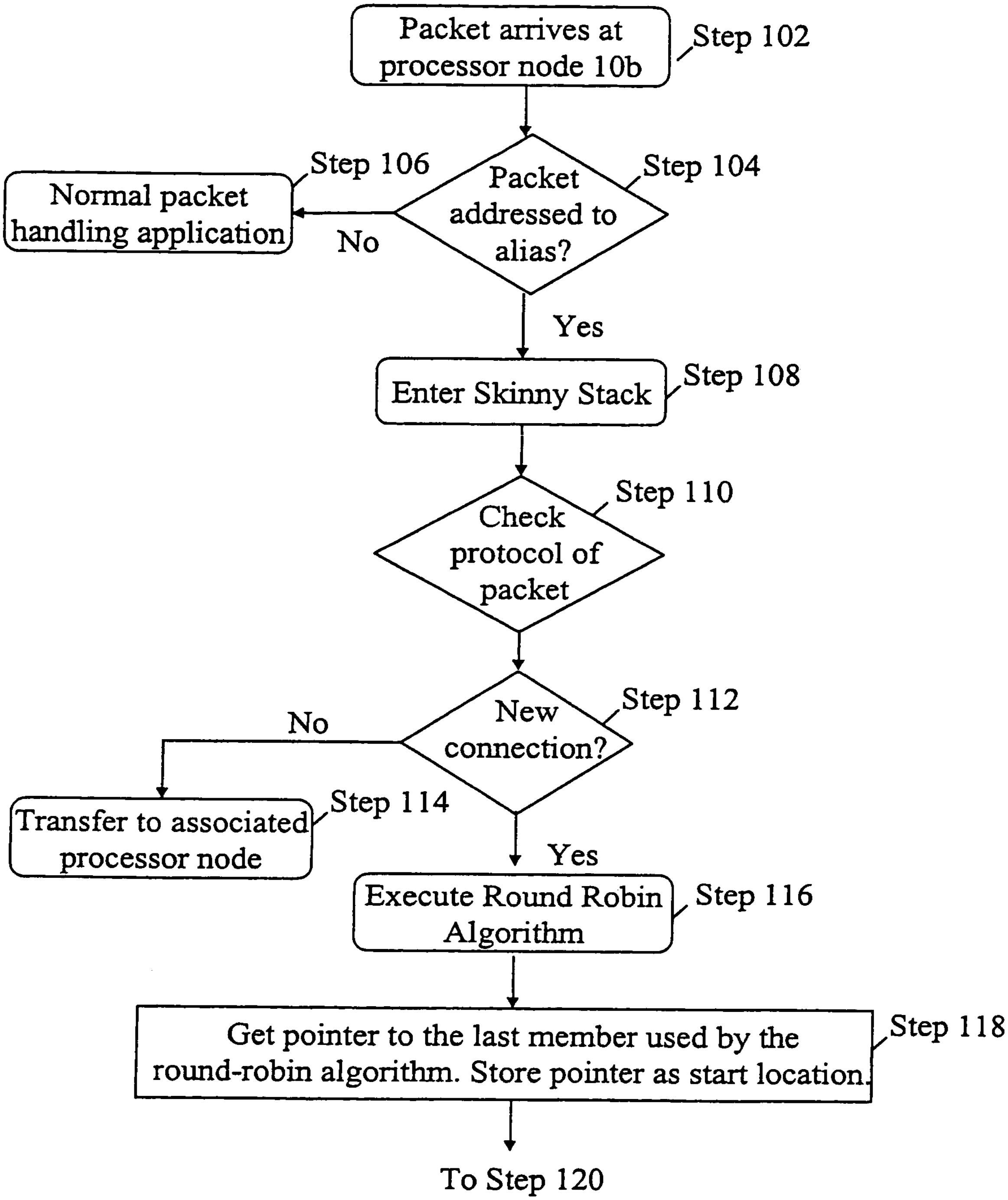

Packet arrives at processor node 10b
Step 102
Step 106
Packet addressed to alias?
Step 104
Normal packet handling application
No
Yes
Enter Skinny Stack
Step 108
Step 110
Check protocol of packet
New connection?
Step 112
No
Transfer to associated processor node
Step 114
Yes
Execute Round Robin Algorithm
Step 116
Get pointer to the last member used by the round-robin algorithm. Store pointer as start location.
Step 118
To Step 120

FIG.5A

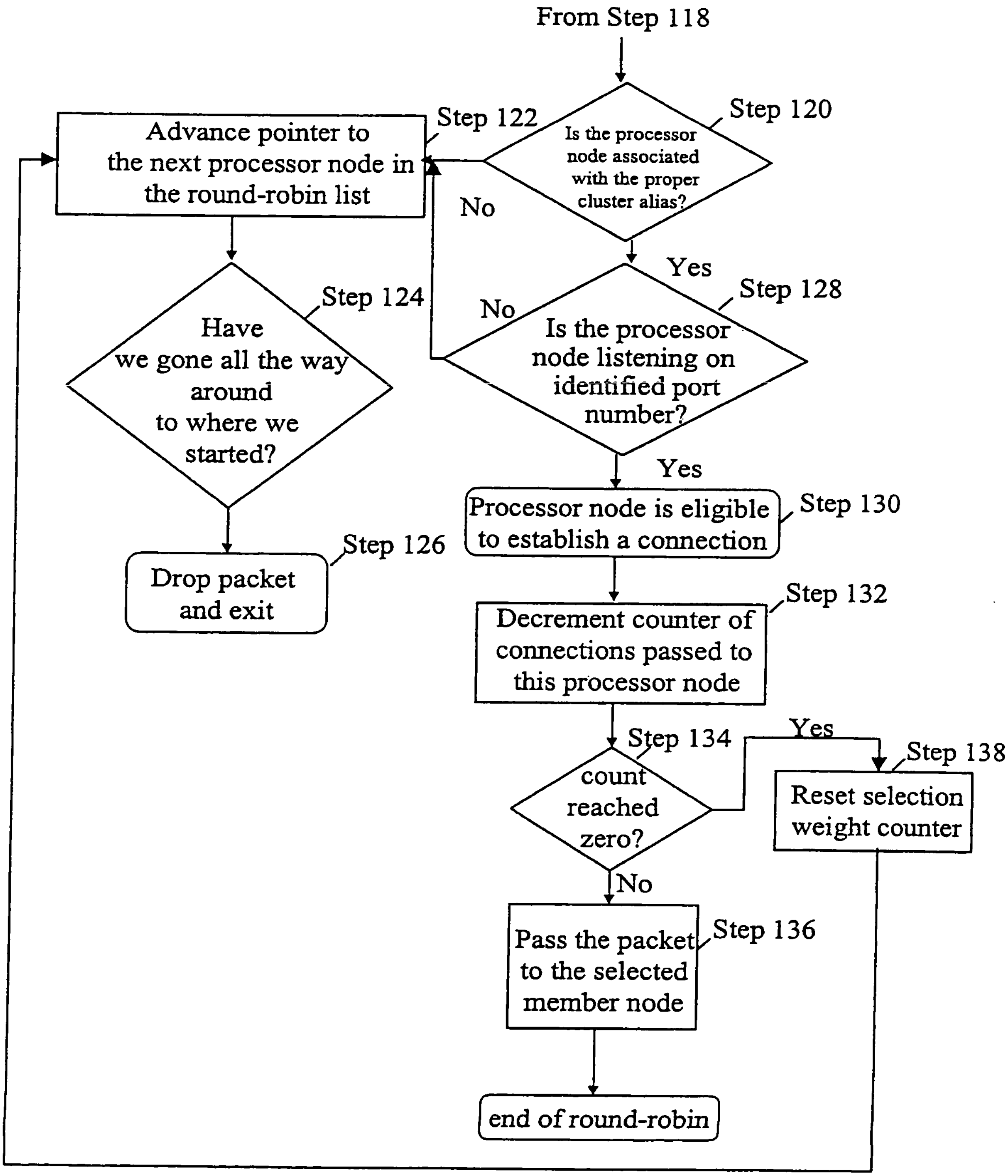

From Step 118
Is the processor node associated with the proper cluster alias?
Step 120
No
Advance pointer to the next processor node in the round-robin list
Step 122
Yes
Is the processor node listening on identified port number?
Step 128
No
Have we gone all the way around to where we started?
Step 124
Drop packet and exit
Step 126
Yes
Processor node is eligible to establish a connection
Step 130
Decrement counter of connections passed to this processor node
Step 132
Step 134
count reached zero?
Yes
Reset selection weight counter
Step 138
No
Pass the packet to the selected member node
Step 136
end of round-robin

FIG.5B

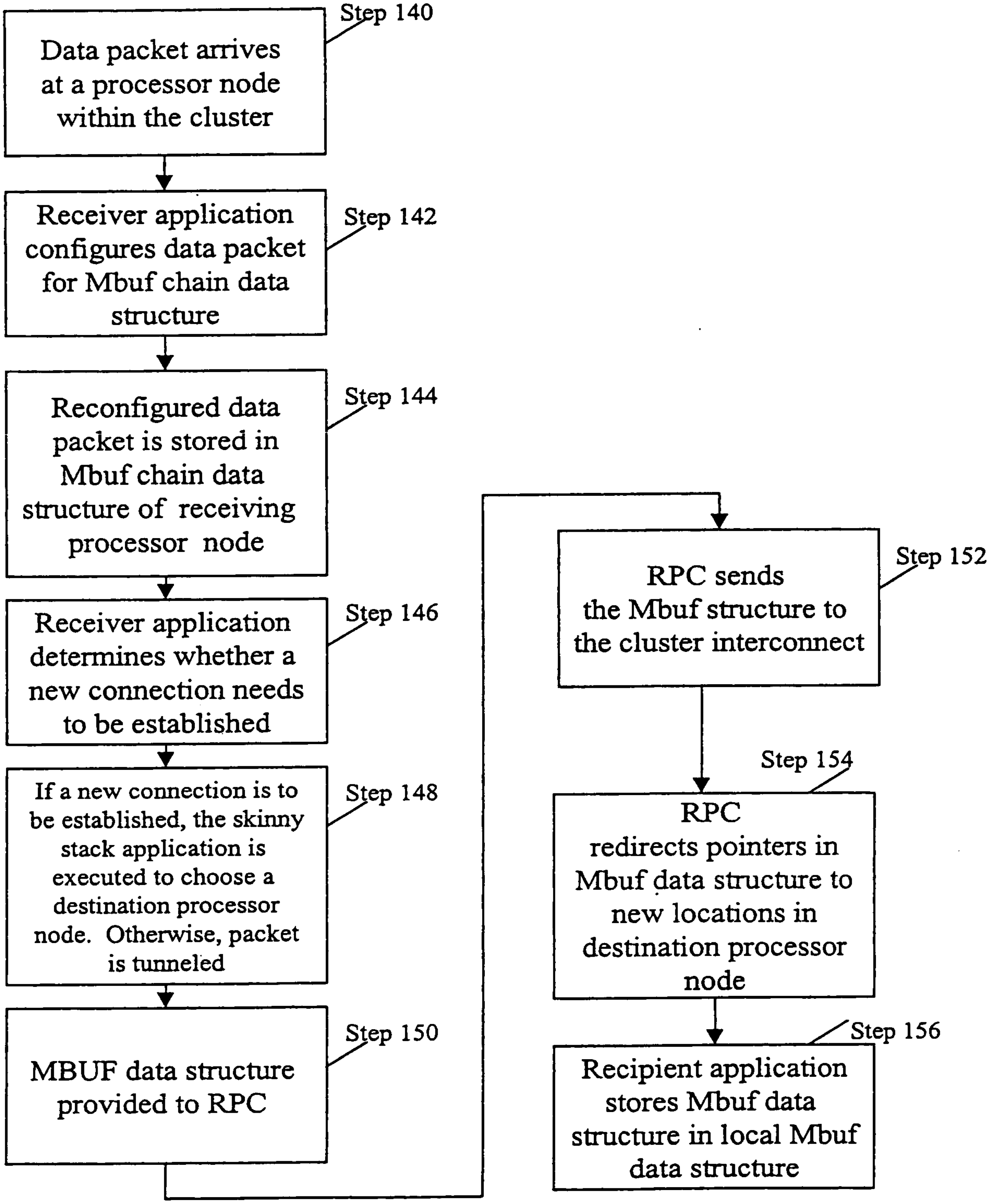

Data packet arrives at a processor node within the cluster
Step 140
Receiver application configures data packet for Mbuf chain data structure
Step 142
Reconfigured data packet is stored in Mbuf chain data structure of receiving processor node
Step 144
Receiver application determines whether a new connection needs to be established
Step 146
If a new connection is to be established, the skinny stack application is executed to choose a destination processor node. Otherwise, packet is tunneled
Step 148
MBUF data structure provided to RPC
Step 150
RPC sends the Mbuf structure to the cluster interconnect
Step 152
Step 154
RPC redirects pointers in Mbuf data structure to new locations in destination processor node
Step 156
Recipient application stores Mbuf data structure in local Mbuf data structure

FIG.6

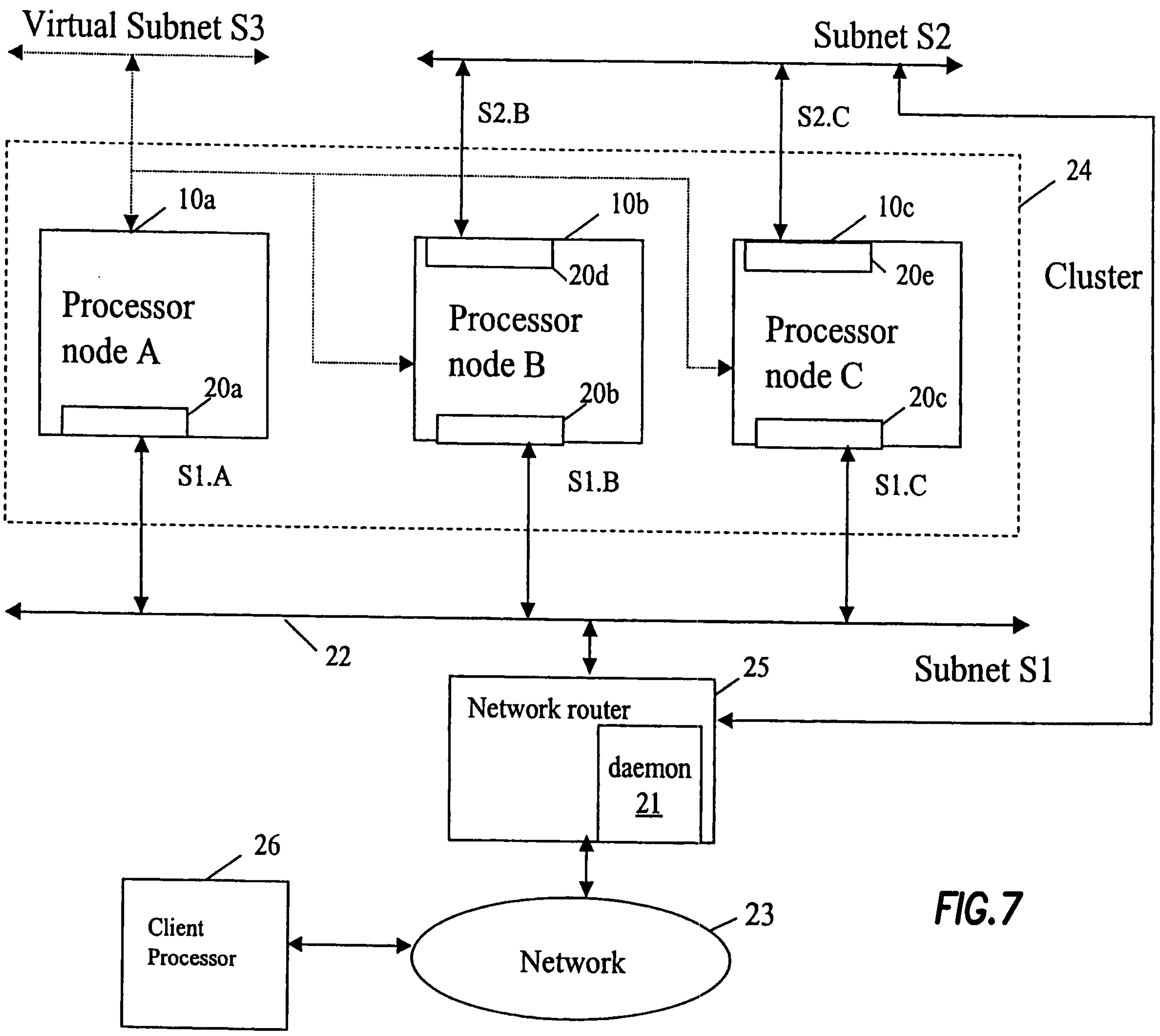

Virtual Subnet S3
Subnet S2
S2.B
S2.C
24
10a
10b
10c
20d
20e
Cluster
Processor node A
Processor node B
Processor node C
20a
20b
20c
S1.A
S1.B
S1.C
22
Subnet S1
25
Network router
daemon
21
26
Client Processor
Network
23

FIG.7

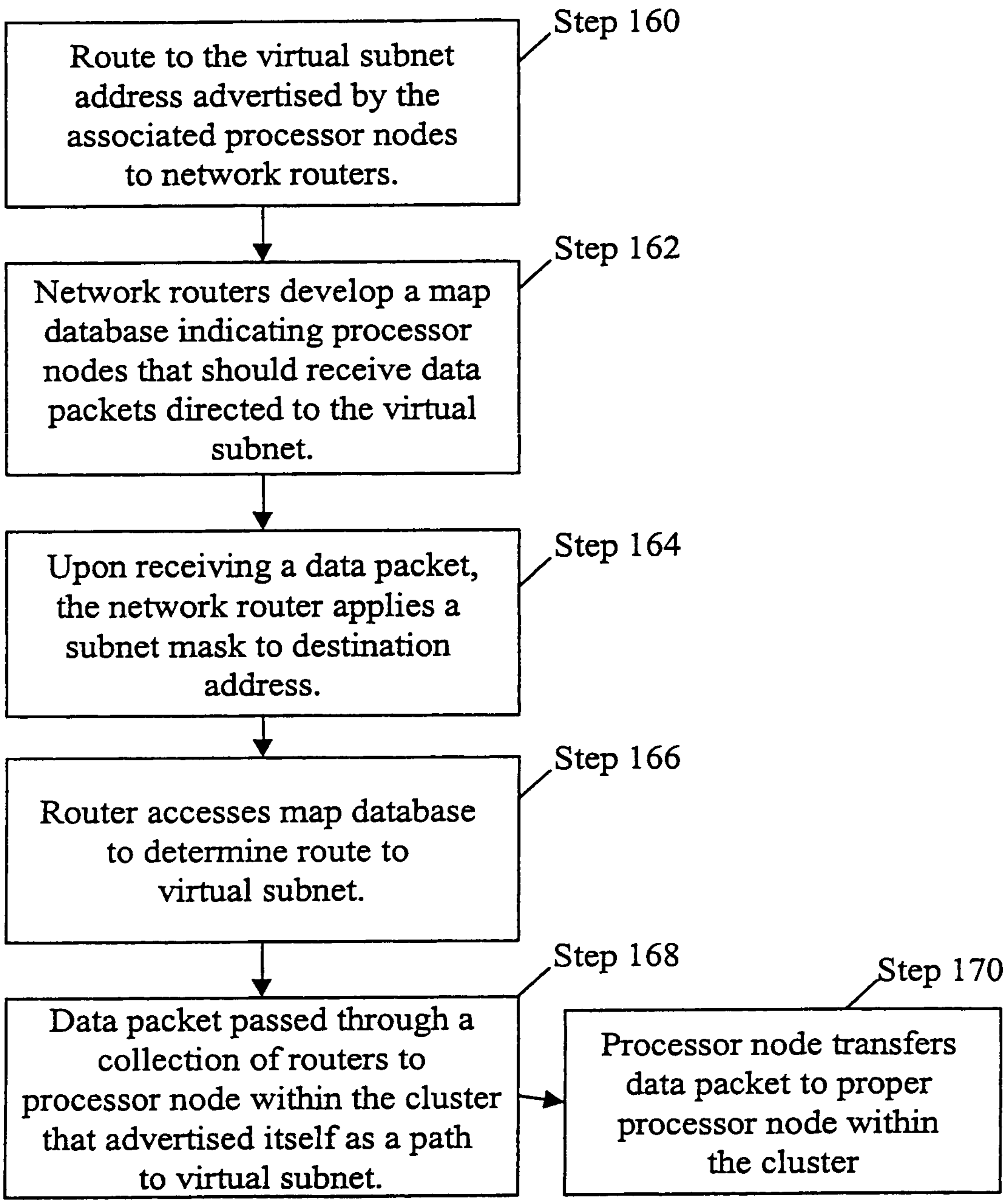

Step 160
Route to the virtual subnet address advertised by the associated processor nodes to network routers.
Step 162
Network routers develop a map database indicating processor nodes that should receive data packets directed to the virtual subnet.
Step 164
Upon receiving a data packet, the network router applies a subnet mask to destination address.
Step 166
Router accesses map database to determine route to virtual subnet.
Step 168
Data packet passed through a collection of routers to processor node within the cluster that advertised itself as a path to virtual subnet.
Step 170
Processor node transfers data packet to proper processor node within the cluster

FIG.8

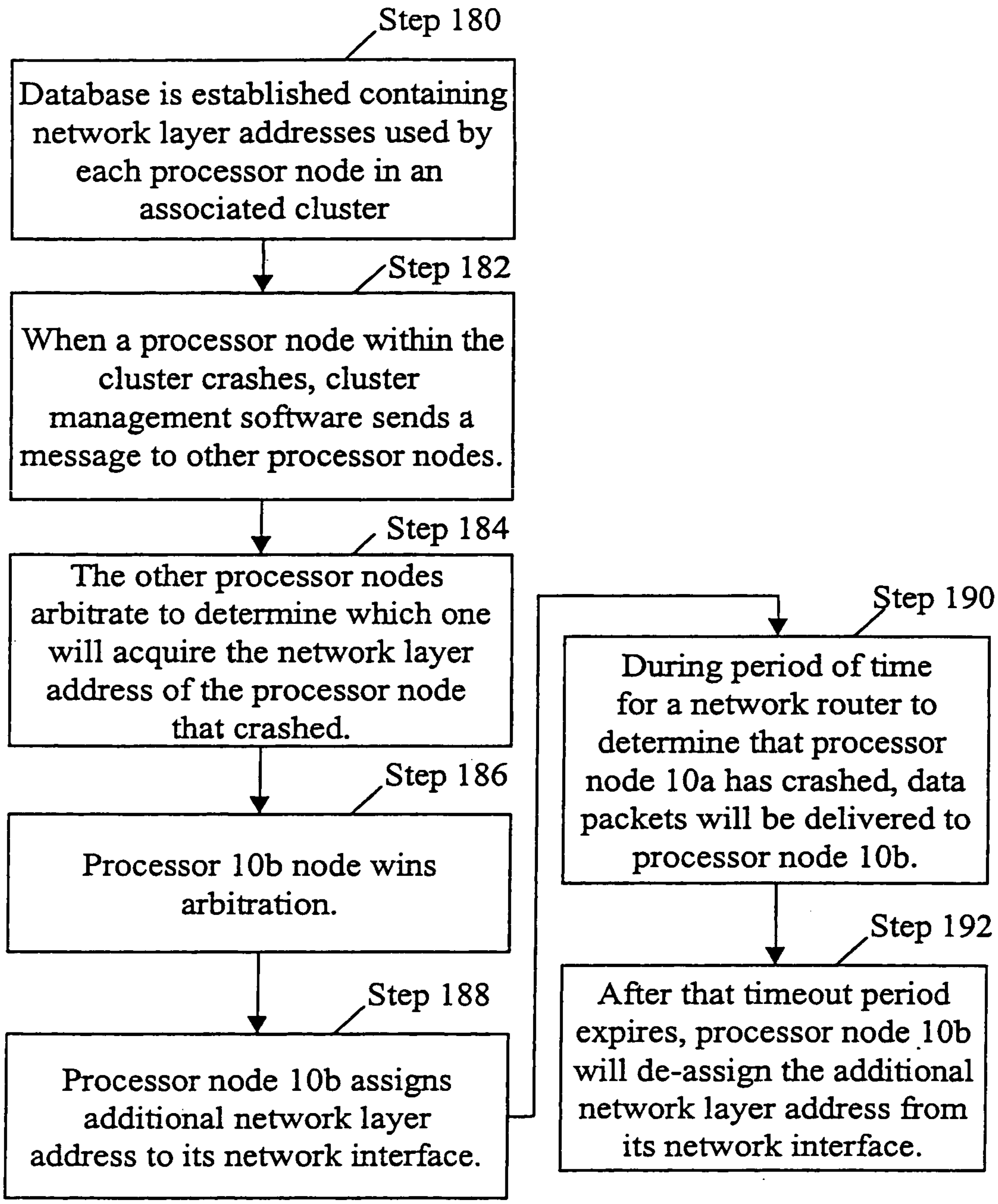

Step 180
Database is established containing network layer addresses used by each processor node in an associated cluster
Step 182
When a processor node within the cluster crashes, cluster management software sends a message to other processor nodes.
Step 184
The other processor nodes arbitrate to determine which one will acquire the network layer address of the processor node that crashed.
Step 186
Processor 10b node wins arbitration.
Step 188
Processor node 10b assigns additional network layer address to its network interface.
Step 190
During period of time for a network router to determine that processor node 10a has crashed, data packets will be delivered to processor node 10b.
Step 192
After that timeout period expires, processor node 10b will de-assign the additional network layer address from its network interface.

FIG.9